CLAIMS

1. An apparatus comprising a rendering engine that defines a rectangular area of pixels that bounds a triangular area of the pixels, and evaluates coordinates associated with the pixels of the rectangular area to selectively render the pixels that fall within the triangular area.

2. The apparatus of claim 1, wherein the rendering engine evaluates the coordinates of the pixels in accordance with a set of linear equations that describe edges of the triangular area.

3. The apparatus of claim 2, wherein the rendering engine computes a coefficient matrix M_C for computing linear coefficients for the set of linear equations, and applies the coefficient matrix M_C to each of the pixels within the rectangular area to determine whether each of the pixels falls within the triangular area.

4. The apparatus of claim 3, wherein the rendering engine applies the coefficient matrix M_C to a current one of the pixels (X_C, Y_C) within the rectangular area to determine whether:

$$M_C \begin{bmatrix} X_C \\ Y_C \\ 1 \end{bmatrix} \leq \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \text{where}$$

the coefficient matrix M_C equals:

$$M_c = \begin{bmatrix} y_1 - y_2 & x_2 - x_1 & x_1 y_2 - x_2 y_1 \\ y_2 - y_0 & x_0 - x_2 & x_2 y_0 - x_0 y_2 \\ y_0 - y_1 & x_1 - x_0 & x_0 y_1 - x_1 y_0 \end{bmatrix} \text{and}$$

vertices $v_0(x_0,y_0)$, $v_1(x_1,y_1)$, and $v_2(x_2,y_2)$ are vertices of the triangular area.

5. The apparatus of claim 1, wherein the rendering engine selectively renders the pixels that fall within the triangular area by computing updated pixel data for those pixels in

accordance with a set of linear equations that describe one or more attributes associated with the triangular area.

6. The apparatus of claim 5, wherein the attribute values comprise at least one of color values and texture values.

7. The apparatus of claim 5, wherein the rendering engine computes a coefficient matrix M^{-1} for computing linear coefficients A, B, C of the set of linear equations, and applies the coefficients A, B, C to each pixel that falls within the triangular area to compute an attribute value for the respective pixel.

8. The apparatus of claim 7, wherein the rendering engine applies the coefficient matrix M^{-1} to compute the linear coefficients A, B, C, for an attribute associated with vertices $v_0(x_0,y_0)$, $v_1(x_1,y_1)$, and $v_2(x_2,y_2)$ of the triangle as:

$$\begin{bmatrix} A \\ B \\ C \end{bmatrix} = M^{-1} \begin{bmatrix} v_0 \\ v_1 \\ v_2 \end{bmatrix},$$

where the coefficient matrix M^{-1} equals:

$$M = \begin{bmatrix} x_0 & y_0 & 1 \\ x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \end{bmatrix}, \text{ and}$$

an attribute value for each pixel (X_c, Y_c) is computed as

$$v = AX_c + BY_c + C.$$

9. The apparatus of claim 1, further comprising a z-buffer storing a set of z-values associated with the pixels, and wherein the rendering engine compares a z-value, z_c, of the current pixel with a corresponding z-value, z_b, of a z-buffer to determine whether each pixel within the rectangular area is visible and selectively renders each pixel of the rectangular area that is visible and that falls within the triangle area.

10. The apparatus of claim 1, further comprising a control unit that issues a command to the rendering engine that specifies vertices of the triangular area.

11. The apparatus of claim 1, wherein the rendering engine comprises:

a vertex buffer for buffering the vertices of the triangular area to be rendered;

a bounding box generator that processes the vertices to compute bounding data that define the dimensions of the rectangular area; and

a rasterizer that processes the bounding data and evaluates coordinates associated with the pixel values of the rectangular area to selectively render the pixels that fall within the triangular area.

12. The apparatus of claim 11, further comprising:

an edge coefficient generator that receives the vertices buffered by the vertex buffer and processes the vertices to compute linear coefficients for a set of linear equations that describe edges of the triangular area, and

an attribute coefficient generator that processes the vertices to compute linear coefficients for a set of linear equations that describe one or more attributes associated with the triangular area, wherein

the rasterizer processes the bounding data and the coefficients in accordance with the sets of linear equations to render the pixels that fall within the triangular area.

13. The apparatus of claim 1, wherein the apparatus comprises a wireless communication device.

14. The apparatus of claim 1, wherein the apparatus comprises an integrated circuit.

15. The apparatus of claim 1, further comprising a cache memory to store at least a portion of the pixels, wherein the cache memory has a block size, and the rendering engine defines the rectangular area as a function of the block size of the cache.

16. A mobile communication device comprising:

a display;

a processor to generate video output data for presentation by the display as a graphical environment; and

a rendering engine that applies a direct evaluation algorithm to render a triangle for the graphical environment, wherein the direct evaluation algorithm applies linear equations to render the triangle without interpolating between edges of the triangle.

17. The mobile communication device of claim 16, wherein the processor issues a command to the rendering engine that defines vertices for the triangle.

18. A method comprising:

computing data that defines a rectangular area of pixels that bounds a triangular area of the pixels;

evaluating coordinates of the pixels of the rectangular area to determine which pixels fall within the triangle area; and

updating pixel data for the pixels that fall within the triangle area to render the triangular area.

19. The method of claim 18, wherein evaluating coordinates comprises evaluating the coordinates of the pixels in accordance with a set of linear equations for computing edges of the triangular area.

20. The method of claim 19, wherein evaluating the coordinates of the pixels comprises:

computing a coefficient matrix M_C for computing linear coefficients for the set of linear equations; and

applying the coefficient matrix M_C to each of the pixels within the rectangular area to determine whether each of the pixels falls within the triangular area.

21. The method of claim 20, wherein applying the coefficient matrix M_C comprises applying the coefficient matrix M_C to a current one of the pixels (X_C, Y_C) within the rectangular area to determine whether:

$$M_C \begin{bmatrix} X_C \\ Y_C \\ 1 \end{bmatrix} \leq \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \text{ where}$$

the coefficient matrix M_C equals:

$$M_c = \begin{bmatrix} y_1 - y_2 & x_2 - x_1 & x_1 y_2 - x_2 y_1 \\ y_2 - y_0 & x_0 - x_2 & x_2 y_0 - x_0 y_2 \\ y_0 - y_1 & x_1 - x_0 & x_0 y_1 - x_1 y_0 \end{bmatrix} \text{and}$$

vertices $v_0(x_0,y_0)$, $v_1(x_1,y_1)$, and $v_2(x_2,y_2)$ are vertices of the triangular area.

22. The method of claim 18, wherein updating pixel data comprises computing pixel data for the pixels of the rectangular area that fall within the triangle area in accordance with a set of linear equations that describe one or more attributes associated with the triangular area.

23. The method of claim 22, wherein the attribute values comprise at least one of color values and texture values.

24. The method of claim 22, wherein updating pixel data comprises:

computing a coefficient matrix M^{-1} for computing linear coefficients of the set of linear equations; and

applying the linear coefficients to each of the pixels that falls within the triangular area to compute an attribute value for each of the pixels.

25. The method of claim 24, wherein applying the coefficient matrix M^{-1} comprises applying the coefficient matrix M^{-1} to compute the linear coefficients A, B, and C, for an attribute associated with vertices $v_0(x_0,y_0)$, $v_1(x_1,y_1)$, and $v_2(x_2,y_2)$ of the triangle as:

$$\begin{bmatrix} A \\ B \\ C \end{bmatrix} = M^{-1} \begin{bmatrix} v_0 \\ v_1 \\ v_2 \end{bmatrix},$$

where the coefficient matrix M^{-1} equals:

$$M = \begin{bmatrix} x_0 & y_0 & 1 \\ x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \end{bmatrix}, \text{ and}$$

an attribute value for each pixel (X_c, Y_c) is computed as

$$v = AX_c + BY_c + C.$$

26. The method of claim 18, further comprising:

determining whether each of the pixels within the rectangular area is visible; and

selectively rendering each of the pixels based on the determination.

27. The method of claim 26, wherein determining whether each of the pixels is visible comprises comparing a z-value, z_c, of the current pixel with a corresponding z-value, z_b, of a z-buffer to determine if $z_c < z_b$.

28. The method of claim 18, wherein computing data comprises computing a first coordinate and a second coordinate that represent opposite corners of the rectangular area.

29. The method of claim 18, wherein

updating pixel data comprises processing the pixel data within a cache memory having a block size; and

computing data to define a rectangular area comprises computing the data to define the rectangular area as a function of the block size of the cache.